



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **09185864 A**(43) Date of publication of application: **15.07.97**

(51) Int. Cl.

**G11B 19/02**(21) Application number: **08285519**(22) Date of filing: **28.10.96**(30) Priority: **30.10.95 JP 07282175**(71) Applicant: **SONY CORP**

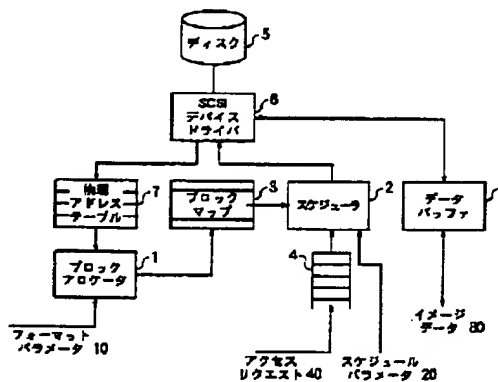
(72) Inventor:  
**TOTSUKA TAKUSHI**  
**KATO YASUNOBU**  
**OYA NOBORU**  
**SHIOTANI HIROYUKI**

**(54) METHOD AND DEVICE FOR CONTROLLING  
 RECORDING DISK ACCESS**

**(57) Abstract:**

**PROBLEM TO BE SOLVED:** To provide a recording disk control method capable of performing recording disk random accessing at a high speed in real time.

**SOLUTION:** A block allocator 1 decides skewing so as to reduce the rotation standby time of a head in the average moving distance of the head, arranges a data block in a specified position on a disk 5 based on the skewing and produces a block map 3. A scheduler 2 performs scheduling for rearranging the order of a plurality of access requests 40 so as to reduce the moving amount of the head and outputs access instruction to a SCSI device driver 6 based on the scheduled order.



COPYRIGHT: (C)1997,JPO

[illegible]

## 【特許請求の範囲】

【請求項1】ヘッドが記録ディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるようにスキューを決定し、  
少なくとも前記決定されたスキューに基づいて、データブロックの記録ディスク上における位置を決定し、  
前記ヘッドによる前記記録ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングし、  
前記スケジューリング結果に基づいて、前記ヘッドによる前記記録ディスクのアクセスを行う記録ディスクアクセス制御方法。

【請求項2】前記データブロックの記録ディスク上における位置の決定は、前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う請求項1に記載の記録ディスクアクセス制御方法。

【請求項3】複数のデータブロックについてスキューとギャップとのそれぞれの組み合わせに関する組み合わせデータを決定し、  
前記記録ディスク上の各データブロックの位置に応じて、前記組み合わせデータを選択的に使用する請求項2

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g + 2m\pi) / \omega \quad (1)$$

## 【数2】

$$L_a = L_t / (N-1)$$

ここでLはシーク距離で単位はシリンダ数、Bcは1シリンダ内に存在するデータブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度(ラジアン/秒)、L<sub>t</sub>はアクセス要求を順に並べた時の両端のアクセス位置間の距離の最大値で単位はシリンダ数、Nは同時に処理するアクセスの個数である。また、mはT<sub>d</sub>(L)がシーク距離Lにおけるシーク時間T<sub>s</sub>(L)を越える範囲で最も小さくなるように選択される。

【請求項7】前記スケジューリングはヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g) / \omega \quad (3)$$

ここでLはシーク距離で単位はシリンダ数、Bcは1シリンダ内に存在するデータブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度(ラジアン/秒)である。

【請求項8】前記記憶ディスク上をヘッドが内周から外周に移動する時に使用する第1の領域と、ヘッドが外周から内周に移動する時に使用する第2の領域とに分割し、

前記データブロックの記録ディスク上における位置の決定は、おのおのの領域においてヘッドの移動方向に基づいて最適なスキューとギャップを決定し、この最適なスキューとギャップに基づいて、前記データブロックの記

に記載の記録ディスクアクセス制御方法。

【請求項4】前記記録ディスクの外側から内側の全領域にわたって前記ギャップが一定であるようにデータブロックの大きさを変化させた請求項1～3のいずれかに記載の記録ディスクアクセス制御方法。

【請求項5】前記記録トラックの半径の差異によって生じるデータブロックのギャップの変化に応じて前記スキューを決定する請求項1～4のいずれかに記載の記録ディスクアクセス制御方法。

【請求項6】前記スケジューリングは、前記記録ヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に前記記録ヘッドに近いものから順に並ぶように複数のディスクアクセスの要求の順序を交換し、

前記データブロックの記録ディスク上における位置の決定は、下記式(1)で与えられるディレイ時間T<sub>d</sub>

(L)と、下記式(2)で与えられる平均シーク距離L<sub>a</sub>付近におけるシーク時間T<sub>s</sub>(L)との差が回転周期と比較して十分に小さくなるように、スキューに加えて、同一のデータブロックの始めと終わりとの間の角度の差異を示すギャップに基づいて行われる請求項1～5のいずれかに記載の記録ディスクアクセス制御方法。

## 【数1】

## (2)

動させた時に会う順番となるように複数のディスクアクセスの要求の順序を交換し、

前記データブロックの記録ディスク上における位置の決定は、下記式(3)で与えられるディレイ時間T<sub>d</sub>

(L)が常にシーク時間T<sub>s</sub>(L)以上であり、かつL<sub>a</sub>付近においてシーク時間T<sub>s</sub>(L)との差が回転周期と比較して十分に小さくなるようにスキューに加えて、ギャップに基づいて行われる請求項1～5のいずれかに記載の記録ディスクアクセス制御方法。

## 【数3】

録ディスク上における位置を決定し、

前記スケジューリングは、ヘッドの移動方向に合わせて前記第1と第2の領域だけを選択的にアクセスする請求項1～7のいずれかに記載の記録ディスクアクセス制御方法。

【請求項9】前記記録ディスク上の第1の領域と第2の領域とはディスクを半径方向に沿って2つ以上の複数の部分に分割し、第1の領域と第2の領域とがともにディスク上の最内周から最外周まで分散するように割り当てる請求項8に記載の記録ディスクアクセス制御方法。

【請求項10】ヘッドが記録ディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるよ

うにスキューを決定するスキュー決定手段と、  
 少なくとも前記決定されたスキューに基づいて、データブロックの記録ディスク上における位置を決定するデータブロック配置手段と、  
 前記ヘッドによる前記記録ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングするスケジューリング手段とを有し、  
 前記ヘッドは、前記スケジューリング結果に基づいて、前記記録ディスクにアクセスを行う記録ディスクアクセス制御装置。

【請求項11】前記データブロック配置手段は、前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う請求項10に記載の記録ディスクアクセス制御装置。

【請求項12】前記データブロック配置手段は、複数のデータブロックについてスキューとギャップとのそれぞれの組み合わせに関する組み合わせデータを決定し、前記記録ディスク上の各データブロックの位置に応じて、前記組み合わせデータを選択的に使用する請求項11に記載の記録ディスクアクセス制御装置。

【請求項13】前記データブロック配置手段は、前記記

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g + 2m\pi) / \omega \quad (4)$$

【数5】

$$L_a = L_t / (N-1) \quad (5)$$

ここでLはシーク距離で単位はシリンダ数、B<sub>c</sub>は1シリンダ内に存在するデータブロック数、θ<sub>s</sub>はスキューで単位はラジアン、θ<sub>g</sub>はギャップで単位はラジアン、ωはディスクの回転速度(ラジアン/秒)、L<sub>t</sub>はアクセス要求を順に並べた時の両端のアクセス位置間の距離の最大値で単位はシリンダ数、Nは同時に処理するアクセスの個数である。また、mはT<sub>d</sub>(L)がシーク距離Lにおけるシーク時間T<sub>s</sub>(L)を越える範囲で最も小さくなるように選択される。

【請求項16】前記スケジューリング手段はヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g) / \omega \quad (6)$$

ここでLはシーク距離で単位はシリンダ数、B<sub>c</sub>は1シリンダ内に存在するデータブロック数、θ<sub>s</sub>はスキューで単位はラジアン、θ<sub>g</sub>はギャップで単位はラジアン、ωはディスクの回転速度(ラジアン/秒)である。

【請求項17】前記記憶ディスク上をヘッドが内周から外周に移動する時に使用する第1の領域と、ヘッドが外周から内周に移動する時に使用する第2の領域とに分割し、

前記データブロック配置手段は、おのおのの領域においてヘッドの移動方向に基づいて最適なスキューとギャップを決定し、この最適なスキューとギャップに基づいて、前記データブロックの記録ディスク上における位置を決定し、

録ディスクの外側から内側の全領域にわたって前記ギャップが一定であるようにデータブロックの大きさを変化させた請求項10～12のいずれかに記載の記録ディスクアクセス制御装置。

【請求項14】前記スキュー決定手段は、前記記録トラックの半径の差異によって生じるデータブロックのギャップの変化に応じて前記スキューを決定する請求項10～13のいずれかに記載の記録ディスクアクセス制御装置。

【請求項15】前記スケジューリング手段は、前記記録ヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に近いものから順に並ぶように複数のディスクアクセスの要求の順序を交換し、前記データブロック配置手段は、下記式(4)で与えられるディレイ時間T<sub>d</sub>(L)と、下記式(5)で与えられる平均シーク距離L<sub>a</sub>付近におけるシーク時間T<sub>s</sub>(L)との差が回転周期と比較して十分に小さくなるように、スキューに加えて、同一のデータブロックの始めと終わりとの間の角度の差異を示すギャップに基づいて行われる請求項10～14のいずれかに記載の記録ディスクアクセス制御装置。

【数4】

移動させた時に出会う順番となるように複数のディスクアクセスの要求の順序を交換し、

前記データブロック配置手段は、下記式(6)で与えられるディレイ時間T<sub>d</sub>(L)が常にシーク時間T<sub>s</sub>(L)以上であり、かつL<sub>a</sub>付近においてシーク時間T<sub>s</sub>(L)との差が回転周期と比較して十分に小さくなるように、スキューに加えて、ギャップに基づいて前記記録ディスク上のデータブロックの位置を決定する請求項10～14のいずれかに記載の記録ディスクアクセス制御装置。

【数6】

前記スケジューリング手段は、ヘッドの移動方向に合わせて前記第1と第2の領域だけを選択的にアクセスする請求項10～16のいずれかに記載の記録ディスクアクセス制御装置。

【請求項18】前記記録ディスク上の第1の領域と第2の領域とはディスクを半径方向に沿って2つ以上の複数の部分に分割し、第1の領域と第2の領域とがともにディスク上の最内周から最外周まで分散するように割り当てた請求項17に記載の記録ディスクアクセス制御装置。

【請求項19】ヘッドがディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるように、データブロックのディスク上における位置を決定

し、

前記ヘッドによる前記ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングし、前記スケジューリング結果に基づいて、前記ヘッドによる前記ディスクのアクセスを行う記録ディスクアクセス制御方法。

【請求項20】前記データブロックのディスク上における位置の決定は、

ヘッドがディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるようにスキューを決定し、

少なくとも前記決定されたスキューに基づいて、前記データブロックのディスク上における位置の決定し、

前記スキューは、隣接するデータブロック間の円周方向における角度の差異を示している請求項19に記載の記録ディスクアクセス制御方法。

【請求項21】前記データブロックのディスク上における位置の決定は、前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う請求項20に記載の記録ディスクアクセス制御方法。

【請求項22】ヘッドがディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるように、データブロックのディスク上における位置を決定する手段と、

前記ヘッドによる前記ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングする手段と、

前記スケジューリング結果に基づいて、前記ヘッドによる前記ディスクのアクセスを行う手段とを有する記録ディスクアクセス制御装置。

【請求項23】前記データブロックのディスク上における位置を決定する手段は、

ヘッドがディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるようにスキューを決定する手段と、

少なくとも前記決定されたスキューに基づいて、前記データブロックのディスク上における位置の決定する手段とを有し、

前記スキューは、隣接するデータブロック間の円周方向における角度の差異を示している請求項22に記載の記録ディスクアクセス制御装置。

【請求項24】前記データブロックのディスク上における位置を決定する手段は、

前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う請求項

23に記載の記録ディスクアクセス制御装置。

【請求項25】ヘッドが記録ディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるようにスキューを決定するスキュー決定手段と、

少なくとも前記決定されたスキューに基づいて、データブロックの記録ディスク上における位置を決定するブロックアロケータと、

前記ヘッドによる前記記録ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングするスケジューラとを有し、

前記ヘッドは、前記スケジューリング結果に基づいて、前記記録ディスクにアクセスを行う記録ディスクアクセス制御装置。

【請求項26】前記ブロックアロケータは、前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う請求項25に記載の記録ディスクアクセス制御装置。

【請求項27】前記ブロックアロケータは、複数のデータブロックについてスキューとギャップとのそれぞれの組み合わせに関する組み合わせデータを決定し、

前記記録ディスク上の各データブロックの位置に応じて、前記組み合わせデータを選択的に使用する請求項26に記載の記録ディスクアクセス制御装置。

【請求項28】前記ブロックアロケータは、前記記録ディスクの外側から内側の全領域にわたって前記ギャップが一定であるようにデータブロックの大きさを変化させた請求項25～27のいずれかに記載の記録ディスクアクセス制御装置。

【請求項29】前記スキュー決定手段は、前記記録トラックの半径の差異によって生じるデータブロックのギャップの変化に応じて前記スキューを決定する請求項25～28のいずれかに記載の記録ディスクアクセス制御装置。

【請求項30】前記スケジューラは、前記記録ヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に近いものから順に並ぶように複数のディスクアクセスの要求の順序を交換し、前記ブロックアロケータは、下記式(7)で与えられるディレイ時間 $T_d(L)$ と、下記式(8)で与えられる平均シーク距離 $L_a$ 付近におけるシーク時間 $T_s(L)$ との差が回転周期と比較して十分に小さくなるように、スキューに加えて、同のデータブロックの始めと終わりとの間の角度の差異を示すギャップに基づいて行われる請求項25～29のいずれかに記載の記録ディスクアクセス制御装置。

【数7】

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g + 2\pi) / \omega \quad (7)$$

【数8】

$$L_a = L_t / (N-1)$$

ここでLはシーク距離で単位はシリンダ数、Bcは1シリンダ内に存在するデータブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度(ラジアン/秒)、 $L_t$ はアクセス要求を順に並べた時の両端のアクセス位置間の距離の最大値で単位はシリンダ数、Nは同時に処理するアクセスの個数である。また、mは $T_d(L)$ がシーク距離Lにおけるシーク時間 $T_s(L)$ を越える範囲で最も小さくなるように選択される。

【請求項31】前記スケジューラはヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移動

$$T_d(L) = (L \cdot Bc \cdot \theta_s + \theta_g) / \omega$$

ここでLはシーク距離で単位はシリンダ数、Bcは1シリンダ内に存在するデータブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度(ラジアン/秒)である。

【請求項32】前記記憶ディスク上をヘッドが内周から外周に移動する時に使用する第1の領域と、ヘッドが外周から内周に移動する時に使用する第2の領域とに分割し、

前記ブロックアロケータは、おのおのの領域においてヘッドの移動方向に基づいて最適なスキューとギャップを決定し、この最適なスキューとギャップに基づいて、前記データブロックの記録ディスク上における位置を決定し、

前記スケジューラは、ヘッドの移動方向に合わせて前記第1と第2の領域だけを選択的にアクセスする請求項25～31のいずれかに記載の記録ディスクアクセス制御装置。

【請求項33】前記記録ディスク上の第1の領域と第2の領域とはディスクを半径方向に沿って2つ以上の複数の部分に分割し、第1の領域と第2の領域とがともにディスク上の最内周から最外周まで分散するように割り当てる請求項32に記載の記録ディスクアクセス制御装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は高速な転送速度とディスク上の不連続な場所のアクセス(ランダム・アクセス)とが必要とされる記録ディスクアクセス制御方法およびその装置に関する。

【0002】

【従来の技術】コンピュータの高速化に伴い、高速でランダムなアクセスが可能なディスク装置はますますその重要性を増している。近年、特に注目されているマルチメディア技術では、デジタルデータとしてディスクに格納した動画像や音声を高速にかつディスク上の互いに離れた場所からアクセスすることが必須である。すなわち、動画像、音声などのマルチメディア・データ用のス

(8)

させた時に出会う順番となるように複数のディスクアクセスの要求の順序を交換し、

前記ブロックアロケータは、下記式(9)で与えられるディレイ時間 $T_d(L)$ が常にシーク時間 $T_s(L)$ 以上であり、かつ $L_a$ 付近においてシーク時間 $T_s(L)$ との差が回転周期と比較して十分に小さくなるように、スキューに加えて、ギャップに基づいて前記記録ディスク上のデータブロックの位置を決定する請求項25～29のいずれかに記載の記録ディスクアクセス制御装置。

【数9】

(9)

ストレージには、高い転送速度とリアルタイム性が必要とされる。高い転送速度は大量の動画像や音声を扱う上で当然必要となる。一方、リアルタイム性とは処理時間の上限が保証されていることを意味する。例えば動画像は毎秒30枚、一定の間隔でつぎつぎに表示されないと動きがごちなくなる。

【0003】また、ディスクの能力が追い付かず音声データが不足すれば音が途切れて不快なノイズを発する。このようにマルチメディアのデータはそれぞれが特定の時刻に用意され使用されないと情報としての価値が激減してしまう。したがってマルチメディア用のストレージでは最悪の場合でもこれだけの時間で処理できる、という上限が保証されていることが重要である。さもないと、たとえ平均的な性能が仕様を満足していても、一時的にデータが間に合わないということが起きかねない。処理時間の最大値が保証されていることをリアルタイム性と呼び、マルチメディア分野では欠かせない機能である。

【0004】従来のコンピュータ用のストレージでは平均性能の向上が第一目標であり、必ずしも最悪値が低く抑えられてはいなかった、つまりストレージ処理時間のばらつきが大きく、この点がマルチメディア用のストレージとは対照的である。しかも、主要なマルチメディアの応用分野(アプリケーション)においては物理的に離れた場所のデータを次々にアクセスすること(ランダム・アクセス)が高速に行なえなくてはならない。例えばビデオ・オン・デマンド(VOD)は大勢の視聴者がそれぞれ好みの番組を好きな時に呼び出して見るシステムであるが、これを実現するには多数の視聴者からの要求を並行して処理し、各視聴者が現在見ている部分のデータを素早く用意しなくてはならない。このため、ディスク上のさまざまな場所に格納されている映画などのソースを高速にたどってまわることが必要になる。

【0005】また、近年ではビデオや映画の編集はテープやフィルムを用いずに、ディスクを使用して行なわれる。テープでは、オーバーライトを避けるために、例えば1時間の番組を作った後で先頭付近に数秒のシーンを

挿入する場合、挿入箇所よりあとのすべての映像を後ろにずらして再度録画しなくてはならない。アナログオーディオカセットテープの高速再生とは異なり、そのようなビデオプログラムの再度の録画にはほぼ番組1本分の時間が必要であり、効率が悪い。しかし、ディスクではランダムアクセスが可能のため、挿入部分をディスク上の別の場所に置いておき、再生時にはその場所にいったんジャンプして挿入された映像をアクセスし、その後でもとの場所に戻り、映像の再生を続けられればよい。この考えをさらに発展させ、番組の各シーン（カット）はディスク上の別々の場所に置いておき、再生時にこれを高速にたどりながらあたかも連続した一本のテープを再生しているかのように見せる技術が知られている。このようにすれば、シーンの入れ換えや長さの変更は、ディスク上のデータをたどる順番を変更するだけで可能なので、編集の効率が極めて高い。これをノン・リニア編集と呼ぶ。ただし、この場合にもディスク上の物理的に離れた場所を高速にたどることが必要である。

【0006】これらの例に見られるように、マルチメディア

$$T_t / (T_t + T_s + T_r)$$

すなわち、ディスク上のデータを端から順に（シーケンシャルに）アクセスする場合と比べて、離れた場所のデータをたどりながら（ランダム・アクセス）する場合にこれはだけの性能の低下を想定しなくてはならない。従ってリアルタイム性（処理時間の上限を定め、常にそれ以下で動作することを保証する）を保ちつつランダム・アクセス時の性能低下を抑えることがマルチメディア用ディスクの課題である。

【0008】

【発明が解決しようとする課題】ところで、ディスクのアクセスのリアルタイム性を保証する方法に関しては近年研究が進められている。例えば、D. Anderson, Y. Osawa, R. Govindan 著の“A File System for Continuous Media”, ACM Transactions on Computer Systems, Vol. 10, No. 4, pp. 311-337, 1992（以下 [Anderson92]）においてはディスクから読み出したデータを一時的に格納するバッファメモリの量と一回のアクセスで読み出すデータ量との関係を最適化することでシステムの性能の向上を計っている。

【0009】しかし、ディスクのアクセスのオーバーヘッドに関しては、解析を容易にするためにシーク時間も回転待ち時間も考えられる最悪の値がランダムアクセスの都度発生すると仮定している。すなわち、シーク時間は最内周から最外周まであるいは最外周から最内周までシークする時の時間が最長シーク時間として採用され、ちょうど一回転分待つための時間が回転待ち時間として採用される。これらの2つの値が加えられてオーバーヘッドが得られる。もちろん、このように仮定すれば処理時間の最悪値の見積もりとして極めて安全であるが、毎回必ずこのような動作をすることは実際にはないので、現実

の応用分野ではディスク上の不連続な場所をたどりながら（これをランダム・アクセスと呼ぶ）高速にデータをアクセスすることが極めて重要であるが、不連続な場所への移動にはシーク時間と呼ばれる、ヘッドが所望のシリンダまで移動するための時間と、回転待ち時間とよばれる、シリンダ内でデータの先頭が現れるまでディスクの回転を待つ時間とが必要である。これらの時間をアクセスのオーバーヘッドと呼ぶ。これが実際にデータをアクセスしている時間に比べて大きいほどディスクからのデータ転送に時間がかかることになり、性能が低下する。

【0007】今、ディスクのシーク時間を $T_s$ 、回転待ち時間を $T_r$ とすれば、ディスクアクセスのオーバーヘッドは $T_s + T_r$ となる。ディスクのヘッドがデータ上に存在し、実際にアクセスをしている時間を $T_t$ とあれば、離れた場所へのヘッドのジャンプが一切ない場合と比べた効率は下記式（10）に示すように低くなる。

【数10】

$$T_t / (T_t + T_s + T_r) \quad (10)$$

にディスクから引き出せる性能と比べて最悪値の見積もりはだいぶ低いものとなり、そのような見積もりは設計データとしての意義が薄い。

【0010】また、V. Rangan, H. Vin 著の“Efficient Storage Techniques for Digital Continuous Multimedia”, IEEE Transactions on knowledge and data engineering, Vol. 5, No. 4, pp. 564-573, 1993（以下 [Rangan93]）においては映像ファイルを複数のセグメントに切り、それぞれのセグメントを別の場所に格納する時、セグメントの長さやセグメント間の間隔をどのように決定すればリアルタイム性を保てるかを検討している。

【0011】しかしここでもセグメント間をジャンプする時（ランダム・アクセス時）には [Anderson92] と同様に毎回最悪のオーバーヘッドが発生すると仮定しているため、同様な問題がある。これらの研究と比較して、最悪値をより低く抑えようとする試みもある。それによって、それだけ高い性能でリアルタイム・ランダムアクセスが保証出来る。N. Reddy, J. Wyllie 著の“Disk scheduling in a multimedia I/O system”, ACM Multimedia 93, pp. 225-233, 1993.（以下 [Reddy93]）、および J. Gemmel, J. Han 他著の“Delay-Sensitive Multimedia on Disk”, IEEE Multimedia 1994, pp. 56-67, 1994.

（以下 [Gemmel94]）、および M. Chen, D. Kandlur, P. Yu 著の“Optimization of the Grouped Sweeping Scheduling (GSS) with Heterogeneous Multimedia Streams”, ACM Multimedia 93, pp. 235-242, 1993.（以下 [Chen93]）はいずれもSCANと呼ばれるヘッドスケジューリング・アルゴリズムを用いてオーバーヘッドを抑えることを試みるものである。

【0012】ヘッドスケジューリングとは、ディスク上の複数の場所のアクセスが必要な場合に、アクセスの順番を工夫することでシーク時間を減少させる方法である。図16に示すSCANアルゴリズムにおいては与えられた複数のI/O要求(#1、#2、...)をディスクの半径方向にソートして順に処理するアルゴリズムであり、I/O要求の到着順(#1、#2、...)に処理したら発生したであろうヘッドの往復動作を防ぐことができ、ひいてはそれぞれのシーク時間を減少させることが出来る。ヘッドスケジューリングのアルゴリズムは多くのものが知られており例えばH.Deitel 著 Operating Systems, Addison Wesley, 1990. に詳しく述べられている。

【0013】[Reddy93]、[Gemmel94]、[Chen93] はいずれもSCANアルゴリズムの使用を前提にしており、このためにシーク時間を抑えることができる。したがってオーバーヘッドの最悪値を低くすることが可能で、[Anderson92]や[Rangan93]と比べて高い性能を保証出来る。しかし、SCANアルゴリズムによって抑えることができるのはシーク時間のみであり、回転待ち時間の低減に関して言及したものはこれまでなかった。[Reddy93]ではゼロ・レイテンシ・アクセス機構と呼ばれる特殊な機能がディスクに存在することを仮定する。ゼロ・レイテンシ・アクセス機構は、ヘッドが所望のトラックに到達した時点で、たとえデータの途中からでもディスクからどんどん読みだし、間に合わなかったデータの先頭部分はあとでディスクが一回転してその部分が戻ってきた時に改めて読む、という方法である。従って、ディスクが1回転すればかならず所望のデータはすべて読み出すことが出来るので、回転待ちとデータのアクセスの合計は最大でも1回転分の時間となる。

【0014】しかし、現実のディスクでこの機構を搭載しているものはほとんどなく、[Reddy93]の仮定は現実的ではないという問題がある。さらに、たとえゼロ・レイテンシ・アクセス機構が搭載されていたとしても、データ長がちょうどトラック長の整数倍でない限り、回転待ちはやはり発生する。例えば、データ長が2.5トラック分の長さであるとき、最初の2トラック分に関しては、ゼロ・レイテンシ・アクセス機構によってシーク後ただちにデータの読み出し・書き込みが開始できるが、最後に残る0.5トラック分については、その部分にヘッドが到達するまでに回転待ちを避けることができない。一方[Gemmel94]は、回転待ちは制御や予測が不可能な点であるとして、常に最大値を加算してオーバーヘッドを見積もる方法を取る。これは安全であるが無駄も多く、そこが問題となる。さらに[Chen93]は回転待ちは無視出来る補正項として扱っているが、これは現実的ではない。例えば最近の高速なディスクでは回転周期が8.3mSであり、一方SCANアルゴリズムを使用すればシーク時間の最大値は6mS程度あるいはそれ以下に抑えることができるので回転待ちは支配的である。さらに、空

気抵抗やモータの消費電力による発熱の面からも、今後回転速度の飛躍的向上を望むことは難しく、回転待ちの減少は解決すべき最大の課題となっている。

【0015】回転待ちの削減が重要であることはコンピュータ用の通常のファイルシステムにおいても指摘されている。S.Ng 著 "Improving Disk Performance Via Latency Reduction", IEEE Transactions on Computers, Vol. 40, No. 1, January 1991, pp. 22-30. (以下 [Ng91]) は回転方向に位相をずらしたデータのコピーを用意する方法でリード時の平均回転待ち時間を削減する方法について述べる。しかし、この方法はデータ量の多いマルチメディア用途への応用は困難である。

【0016】本発明は以上のような状況に鑑みてなされたものであり、シーク時間と回転待ち時間の両方を低く抑えることで、ランダム・アクセスをリアルタイム性を保ちつつ高速に実現できる記録ディスクアクセス制御方法およびその装置を提供することを目的とする。このような、記録ディスクアクセス制御方法およびその装置は需要の増しつつあるマルチメディア・データ用のストレージとして好適である。

#### 【0017】

【課題を解決するための手段】上述した従来技術の問題点を解決し、上述した目的を達成するために、本発明の記録ディスクアクセス制御方法は、ヘッドが記録ディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるようにスキューを決定し、少なくとも前記決定されたスキューに基づいて、データブロックの記録ディスク上における位置を決定し、前記ヘッドによる前記記録ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングし、前記スケジューリング結果に基づいて、前記ヘッドによる前記記録ディスクのアクセスを行う。

【0018】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記データブロックの記録ディスク上における位置の決定は、前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う。

【0019】また、本発明の記録ディスクアクセス制御方法は、好ましくは、複数のデータブロックについてスキューとギャップとのそれぞれの組み合わせに関する組み合わせデータを決定し、前記記録ディスク上の各データブロックの位置に応じて、前記組み合わせデータを選択的に使用する。

【0020】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記記録ディスクの外側から内側の全領域にわたって前記ギャップが一定であるようにデータブロックの大きさを変化させる。

【0021】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記記録トラックの半径の差異に



よって生じるデータブロックのギャップの変化に応じて前記スキューを決定する。

【0022】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記スケジューリングは、前記記録ヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に前記記録ヘッドに近いものから順に並ぶように複数のディスクアクセスの要求の順序を交換し、前記データブロックの記録ディスク上における位置の決定は、下記式(11)で与えられるデ

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g + 2m\pi) / \omega \quad (11)$$

【0024】

$$L_a = L_t / (N-1)$$

ここでLはシーク距離で単位はシリンダ数、B<sub>c</sub>は1シリンダ内に存在するデータブロック数、θ<sub>s</sub>はスキューで単位はラジアン、θ<sub>g</sub>はギャップで単位はラジアン、ωはディスクの回転速度(ラジアン/秒)、L<sub>t</sub>はアクセス要求を順に並べた時の両端のアクセス位置間の距離の最大値で単位はシリンダ数、Nは同時に処理するアクセスの個数である。また、mはT<sub>d</sub>(L)がシーク距離Lにおけるシーク時間T<sub>s</sub>(L)を越える範囲で最も小さくなるように選択される。

【0025】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記スケジューリングはヘッドを

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g) / \omega \quad (13)$$

ここでLはシーク距離で単位はシリンダ数、B<sub>c</sub>は1シリンダ内に存在するデータブロック数、θ<sub>s</sub>はスキューで単位はラジアン、θ<sub>g</sub>はギャップで単位はラジアン、ωはディスクの回転速度(ラジアン/秒)である。

【0027】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記記憶ディスク上をヘッドが内周から外周に移動する時に使用する第1の領域と、ヘッドが外周から内周に移動する時に使用する第2の領域とに分割し、前記データブロックの記録ディスク上における位置の決定は、おのおのの領域においてヘッドの移動方向に基づいて最適なスキューとギャップを決定し、この最適なスキューとギャップに基づいて、前記データブロックの記録ディスク上における位置を決定し、前記スケジューリングは、ヘッドの移動方向に合わせて前記第1と第2の領域だけを選択的にアクセスする。

【0028】また、本発明の記録ディスクアクセス制御方法は、好ましくは、前記記録ディスク上の第1の領域と第2の領域とはディスクを半径方向に沿って2つ以上の複数の部分に分割し、第1の領域と第2の領域とがともにディスク上の最内周から最外周まで分散するように割り当てる。

【0029】また、本発明の記録ディスクアクセス制御装置は、ヘッドが記録ディスクに対してアクセスするときの平均移動距離での回転待ち時間が短くなるようにスキューを決定するスキュー決定手段と、少なくとも前記決定されたスキューに基づいて、データブロックの記録

イレイ時間T<sub>d</sub>(L)と、下記式(12)で与えられる平均シーク距離L<sub>a</sub>付近におけるシーク時間T<sub>s</sub>(L)との差が回転周期と比較して十分に小さくなるように、スキューに加えて、同一のデータブロックの始めと終わりとの間の角度の差異を示すギャップに基づいて行われる。

【0023】

【数11】

【数12】

(12)

現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に会う順番となるように複数のディスクアクセスの要求の順序を交換し、前記データブロックの記録ディスク上における位置の決定は、下記式(13)で与えられるディレイ時間T<sub>d</sub>(L)が常にシーク時間T<sub>s</sub>(L)以上であり、かつL<sub>a</sub>付近においてシーク時間T<sub>s</sub>(L)との差が回転周期と比較して十分に小さくなるようにスキューに加えて、ギャップに基づいて行われる。

【0026】

【数13】

ディスク上における位置を決定するデータブロック配置手段と、前記ヘッドによる前記記録ディスクに対するアクセス時に、前記ヘッドの移動量が少なくなるように、入力した複数のディスクアクセス要求の順番をスケジューリングするスケジューリング手段とを有し、前記ヘッドは、前記スケジューリング結果に基づいて、前記記録ディスクにアクセスを行う。

【0030】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記データブロック配置手段は、前記スキューに加えて、同一データブロック内での先頭と末尾との角度差を示すギャップに基づいて行う。

【0031】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記データブロック配置手段は、複数のデータブロックについてスキューとギャップとのそれぞれの組み合わせに関する組み合わせデータを決定し、前記記録ディスク上の各データブロックの位置に応じて、前記組み合わせデータを選択的に使用する。

【0032】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記データブロック配置手段は、前記記録ディスクの外側から内側の全領域にわたって前記ギャップが一定であるようにデータブロックの大きさを变化させる。

【0033】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記スキュー決定手段は、前記記録トラックの半径の差異によって生じるデータブロックのギャップの変化に応じて前記スキューを決定する。

【0034】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記スケジュール手段は、前記記録ヘッドを現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に近いものから順に並ぶように複数のディスクアクセスの要求の順序を交換し、前記データブロック配置手段は、下記式(14)で与えられるディレイ時間 $T_d(L)$ と、下記式(15)で与

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g + 2m\pi) / \omega \quad (14)$$

【0036】

$$L_a = L_t / (N-1) \quad (15)$$

ここで $L$ はシーク距離で単位はシリンダ数、 $B_c$ は1シリンダ内に存在するデータブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度(ラジアン/秒)、 $L_t$ はアクセス要求を順に並べた時の両端のアクセス位置間の距離の最大値で単位はシリンダ数、 $N$ は同時に処理するアクセスの個数である。また、 $m$ は $T_d(L)$ がシーク距離 $L$ におけるシーク時間 $T_s(L)$ を越える範囲で最も小さくなるように選択される。

【0037】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記スケジュール手段はヘッドを

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g) / \omega \quad (16)$$

ここで $L$ はシーク距離で単位はシリンダ数、 $B_c$ は1シリンダ内に存在するデータブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度(ラジアン/秒)である。

【0039】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記記憶ディスク上をヘッドが内周から外周に移動する時に使用する第1の領域と、ヘッドが外周から内周に移動する時に使用する第2の領域とに分割し、前記データブロック配置手段は、おのおのの領域においてヘッドの移動方向に基づいて最適なスキューとギャップを決定し、この最適なスキューとギャップに基づいて、前記データブロックの記録ディスク上における位置を決定し、前記スケジュール手段は、ヘッドの移動方向に合わせて前記第1と第2の領域だけを選択的にアクセスする。

【0040】また、本発明の記録ディスクアクセス制御装置は、好ましくは、前記記録ディスク上の第1の領域と第2の領域とはディスクを半径方向に沿って2つ以上の複数の部分に分割し、第1の領域と第2の領域とがともにディスク上の最内周から最外周まで分散するように割り当てる。

【0041】本発明の記録ディスクアクセス制御方法およびその装置では、データブロックの配置を行なう際に円周方向にずらしながら(スキューを与えて)配置するため、シーク動作中に所望のデータの先頭がヘッドの位置を通り過ぎないように制御できる。このため、所望のデータの先頭が再び戻ってくるまでの時間を待つ必要がなく、シーク時間と回転待ち時間の両方を低く抑え、ラ

えられる平均シーク距離 $L_a$ 付近におけるシーク時間 $T_s(L)$ との差が回転周期と比較して十分に小さくなるように、スキューに加えて、同一のデータブロックの始めと終わりとの間の角度の差異を示すギャップに基づいて行われる。

【0035】

【数14】

【数15】

現在位置から前記記録ディスクの内周もしくは外周に向かって移動させた時に会う順番となるように複数のディスクアクセスの要求の順序を交換し、前記データブロック配置手段は、下記式(16)で与えられるディレイ時間 $T_d(L)$ が常にシーク時間 $T_s(L)$ 以上であり、かつ $L_a$ 付近においてシーク時間 $T_s(L)$ との差が回転周期と比較して十分に小さくなるように、スキューに加えて、ギャップに基づいて前記記録ディスク上のデータブロックの位置を決定する。

【0038】

【数16】

ンダム・アクセスをリアルタイム性を保ちつつ高速に実現できる。また、本発明の記録ディスクアクセス制御方法およびその装置では、ヘッドが外周から内周に移動している時に使用する領域と、内周から外周に移動している時に使用する領域とでスキューを変化させることでヘッドの移動方向に関わらず最適なスキューを与えることができる。

【0042】

【発明の実施の形態】以下、本発明の実施形態に係わる記録ディスクアクセス制御方法およびその装置について説明する。

#### 第1実施形態

図1は、本実施形態に係わる記録ディスクアクセス制御装置の構成図である。本実施形態に係わる記録ディスクアクセス制御装置は、例えば、コンピュータにおいてソフトウェアを動作させることによって実現され、図1に示す各ブロックは主要なプログラムモジュールあるいは主要なデータ構造を表している。ブロックアロケータ

(配置手段)1はディスク5にデータをどのように配置するかを、与えられたフォーマットパラメータ10をもとに決定する。ディスク5としては、例えば、磁気ディスク、光磁気(MO)ディスクあるいはハードディスク(HDD)などが用いられる。

【0043】このフォーマットパラメータ10は第1のデータのブロックの大きさ、SCANスケジュールを行なう時の平均のヘッド移動距離 $L_a$ 、使用するドライブのシーク時間関数 $T_s(L)$ 、ディスク5の物理的フォーマットを含む。平均のヘッド移動距離 $L_a$ は、ディスクの

総シリンダ数 $L_t$ と、1回のスキャンで処理するアクセスの個数 $N$ から下記式(17)で与えられる。

$$L_a = L_t / (N-1)$$

【0044】ドライブのシーク時間 $T_s(L)$ はシーク距離 $L$ (シリンダ数)の関数であり、その値は使用するディスクドライブの機械的特性によって決まる。図4にその例を示す。何個のアクセスリクエストをまとめてスキャンで処理するかは、このディスクを使用するアプリケーションの性格、そこで求められる性能、使用できるバッファメモリの量などから決定する。まとめてスキャンするアクセスリクエストの数 $N$ は大きいほどディスクのランダムアクセス性能が向上するが、必要なバッファメモリの量が増加するとともに、応答時間が増加するという副作用がある。

【0045】これらのフォーマットパラメータ10は、システム全体を管理するプログラム(図示しない)を処理する中央プロセッサユニット(CPU)が決定してブロックアロケータ1に与える。本実施形態をノンリニア編集用に応用した実施例では $N=10$ 、 $L_a=300$ をブロックアロケータ1に与えた。また、データ1ブロックの大きさは画像データ1枚分であり、放送局などで使用されるCCIR-601というフォーマットの場合は約700KB(キロ・バイト)である。もちろんこの数値は用途やさまざまな要求仕様にあわせて自由に設定可能である。

【0046】ブロックアロケータ1は与えられたフォーマットパラメータ10に基づいて、各ブロックをディスク5上のどこに置くか決定する。この例では1ブロックは画像1フレームであるが、もちろんMPEGなどの圧縮画像を適当な長さ分割したものでも、またオーディオデータでも、基本的な考え方は同様である。ディスク5は通常セクタと呼ばれる領域ごとにアクセスすることができる。一つのセクタは通常512B(バイト)から4KB程度の大きさで、これを円周上に並べたドーナツ型の領域をトラック、さらに複数枚重ねた磁気メディアの同じトラックを集めた円筒状の領域をシリンダとよぶ。

【0047】画像や音声データの1ブロックは通常1セクタよりも大きいので、ブロックアロケータ1はそれぞれのブロックに対して複数のセクタを割り当てる。図5に磁気メディアが1枚の場合の簡単な例を示す。図5において斜線によって示した部分、すなわちトラック

「1」の一周全部とトラック「2」のセクタ「0」～「6」とが1ブロックに対応する。この例ではメディアは1枚なのでトラックとシリンダが同じ意味であったが、磁気メディアが複数枚存在するディスクドライブの場合は同じシリンダ内を全部使い切ったら隣接シリンダを使用するように割り当てを行なう。

【0048】セクタへの割り当てを全部のブロックについて行ない、その結果をブロックマップ3に記入してブ

【数17】

(17)

ロックアロケータ1の役割は終了する。ブロックアロケータ1はセクタの位置をシリンダ番号、メディア番号(何枚目のメディアか)、セクタ番号の組で指定する。しかし、近年最も普及しているSCSI(ANSI Small Computer System Interface)仕様のドライブではドライブ内のすべてのセクタに通し番号(論理セクタ番号)を与え、これを用いてデータをアクセスするようになっている。尚、SCSIでは、論理セクタのことを論理ブロックアドレスと呼ぶが、画像や音声ブロックとの混同を避けるために、論理セクタという言葉を用いる。そのため、物理アドレステーブル7に予めドライブが定める論理セクタ番号と物理的なアドレスすなわち(シリンダ番号、メディア番号、セクタ番号)との対応を格納しておき、ブロックアロケータ1は物理アドレステーブル7を参照して、所望の物理アドレスをSCSIの定める論理セクタ番号に変換し、これをもブロックマップ3に記入する。

【0049】図6にブロックマップの例を示す。ブロック「0」に対応する情報は図5の斜線部と対応している。一方、スケジューラ2は、以下のように動作する。まず、システム全体を管理するプログラム(図示しない)がスケジューラパラメータ20を決定し、スケジューラパラメータ20をスケジューラ2に与える。スケジューラパラメータ20には何個のアクセスリクエストをまとめて1回のスキャンで処理するかを示す定数 $N$ が含まれる。

【0050】さて、オペレータがディスク5に記録された動画の再生開始を指示すると、図示しない適切な制御プログラムを持つCPUが、動画を構成する画像の1枚1枚について、それが格納されているブロックへのアクセスリクエストを出す。これらのアクセスリクエスト40はアクセスリクエストバッファ4に格納される。スケジューラ2は、アクセスリクエストバッファ4に格納されたアクセスリクエスト40を先に到着したものから順に $N$ 個ずつ取り出し、ブロックマップ3を参照してこれらのリクエストと対応するデータのディスク5上の位置を求め、ヘッドの移動量が最小となるようにアクセスリクエストの順番を入れ換えてディスク5にアクセスの指示を出す。アクセスの指示はディスクドライブの外部インタフェースに合わせるため、SCSIデバイスドライバ6でSCSIのプロトコルに変換されディスク5に伝えられる。ディスク5から読み出されたデータはデータバッファ8にいったん格納され、さらに装置のビデオインタフェースに転送される。

【0051】また、オペレータが、動画のディスク5への記録開始を指示すると、図示しない適切な制御プログラムを持つCPUは、各イメージ毎に、その動画を

構成するイメージが記録されたブロックにアクセス要求を発する。これらのアクセスリクエスト40は、アクセスリクエストバッファ4に記録される。それと同時に、動画像を構成するイメージデータ80は、ビデオインターフェース（図示しない）からデータバッファ8に転送される。そのイメージデータ80は、データバッファ8に一時的に記録される。スケジューラ2は、アクセスリクエストバッファ4に記録されたN個のアクセス要求40を、先に到着したものから順に取り出す。次に、スケジューラ2はブロックマップ3を参照しながら、これらのリクエストに対応したデータのディスク5上の位置を検出する。さらに、スケジューラ2は、ヘッドの総動作量が最小になるように、アクセスリクエストの順番を替え、ディスク5にアクセスするための指示を生成する。そのアクセス指示は、ディスクドライブの外部インターフェースに合うように生成され、SCSIデバイスドライバ6によってSCSIプロトコルに変換され、ディスク5に転送される。

【0052】以下、ブロックアロケータ1の動作について詳細に説明する。図1におけるフォーマットパラメータ10として、1ブロックの大きさ、SCANスケジュールを行なう時の平均のヘッド移動距離 $L_a$ 、使用するドライブのシーク時間関数 $T_s(L)$ 、ディスク5の物理的フォーマット（シリンダ数、1トラック内のセクタ数、シリンダを構成するメディアの枚数）が与えられると、

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g + 2 \cdot m \cdot \pi) / \omega \quad (18)$$

【0057】ここで、 $L$ はシーク距離で単位はシリンダ数、 $B_c$ は1シリンダ内に存在するブロック数、 $\theta_s$ はスキューで単位はラジアン、 $\theta_g$ はギャップで単位はラジアン、 $\omega$ はディスクの回転速度（ラジアン/秒）、 $m$ は $T_d(L)$ が正となるような任意の整数である。

【0058】図7は、上記式(18)の意味をディスク上で説明するものである。図7においていまちょうどブロック「0」のアクセスが終了したとする。ヘッドは中心から見て角度 $\theta_0$ の方向にあるものとする。いま、同じブロック「0」を再度アクセスすることを考えるとギャップ $\theta_g$ の分だけディスクが回転するのを待たないといけないから $\theta_g/\omega$ の待ち時間が発生する。

【0059】また、ブロック「n」の先頭に関してはブロック「0」のギャップ $\theta_g$ とnブロック分のスキュー( $n\theta_s$ )との和の角度だけディスクの回転を待たなくてはならない。これには、 $n\theta_s/\omega$ の時間を要する。ディスクは回転しているからこうして得られた時間に回転周期( $T_{rot}$ )の整数倍を加えた時刻にもデータの先頭はヘッドの存在する位置に到達する。nブロック移動する時、シリンダ数では $n/B_c$ シリンダの移動に相当するので、横軸をシリンダ数、縦軸を先頭が到着するまでの待ち時間としてグラフを描くと図8のようにな

$$T_d(L) = (L \cdot B_c \cdot \theta_s + \theta_g) / \omega \quad (19)$$

【0062】続くステップS3-2、S3-3およびS

ブロックアロケータ1は各ブロックのディスク5上での位置の決定を図2に示すステップS1~S5の手順で行なう。

【0053】ステップS1で1シリンダ内に画像のブロックが何個存在するか( $B_c$ )を計算する。1シリンダ内のセクタの総数はトラック内のセクタ数にメディアの枚数を乗じたものである。1ブロックを格納するために必要なセクタ数でこれを割れば、 $B_c$ が求まる。

【0054】ステップS2でギャップ $\theta_g$ を求める。ギャップはブロックの先頭セクタと最終セクタとの角度差である。例として、図5に斜線で示したブロックの場合は先頭がトラック「1」のセクタ「0」、末尾はトラック「2」のセクタ「6」であるから、ギャップ $\theta_g$ は円周の $5/12$ 、すなわち $5\pi/6$ ラジアンである。

【0055】以上のデータをもとにステップS3でスキュー $\theta_s$ を求める。ここで、スキュー $\theta_s$ とは、ディスク5上の隣接するブロックの先頭同士の円周方向における角度差をいう。まず、あるブロックを読み終わった時点でのヘッドの位置を起点として、そこからLシリンダ移動した位置においてデータの先頭が円周方向の同じ角度に達するまでの時間 $T_d(L)$ を式で表すと、下記式(18)のようになる。

【0056】

【数18】

る。スキューを大きくするほど直線群の傾きは大きくなる。なお、以上の議論では中心から見たヘッドの円周方向の位置（角度）が、中心からの距離によらず一定であるとした。実際にはヘッドの機構によっては正確に位置が一定でない場合もあるが、その影響は十分に小さいので通常は無視出来る。

【0060】図8にしめすように、上記式(18)によって各シリンダにおいてブロックの先頭がヘッド下に到達するまでの時間が得られる。ただし、この時間内にヘッドが所望のシリンダまで移動していないといけなから、待ち時間はシーク後に、最初にブロックの先頭が現れるまでの時間となる。これがシークと回転待ちの両方を考慮したオーバヘッド $T_d(L)$ である。図9にその例を示し、下記式(19)にその定義をしめす。実際のオーバヘッド（待ち時間）は図9において太線で示した。なおシーク時間の関数は点線でしめしてある。図中、 $T_{rot}$ は1回転周期である。

【0061】ステップS3-1では、式(18)や図8における $m=0$ の直線の方程式、すなわち、下記式(19)で示される方程式を求める。

【数19】

3-4は、この直線が常にシーク時間の関数 $T_s(L)$

よりも上で(大きく)かつシーク時間の関数にほぼ接するようにスキュー $\theta_s$ を選択するステップである。図9の $m=0$ の直線はこうして選択された $\theta_s$ を用いて得られたものである。

【0063】ステップS4、S5は以上のようにして得られたスキューとギャップを用いて、ディスク全域にわたって各ブロックのディスク上位置を決定するものである。ステップS4でまず物理アドレスのポインタを(0/0/0)に初期化する。次のステップS5は全てのブロックについて繰り返すループであり、ループの内部ではまずステップS5-1において物理アドレス表を参照して物理アドレスから論理セクタ番号を得、ステップS5-2においてこれと物理アドレスなどの情報をブロックマップ3に書き込む。ブロックマップ3の内容を図6に示す。

【0064】ここまでの処理が終わったら、次のブロックの処理の準備として物理アドレスのポインタを進める。物理アドレスのポインタ $P_{pa}$ はステップS5-3において更新する。ステップS5-4では、全てのブロックについて、ステップS5-1～S5-3までの処理を行ったか否かを判断し、行っていないければ、処理を行っていないブロックについてステップS5-1～S5-3までの処理を行う。ここで、第N番目のブロックのディスク上の配置は、(1)割り当てたブロックより後方で、(2)かつ第0ブロックの先頭との角度差が $N\theta_s$ に最も近いセクタを先頭とする領域である。

【0065】次に、本実施形態に係わる記録ディスクアクセス制御装置における最悪のオーバーヘッドについて説明する。一般にアクセスリクエスト40はディスク5上のあらゆる場所に対して発生する。一回のスキャンで処理する場所は図10(a)や(b)に示すように分布に偏りがあったり、逆に(c)に示すように均等に分布したりする。図10の例では6個のアクセスリクエスト40の間を移動するために5回のランダムアクセスとそれに伴うオーバーヘッドが発生する。そのひとつずつについて、図9の太線にしめすオーバーヘッドが発生する。この5回のランダムアクセスに対するオーバーヘッドの総和が最悪になるのは、オーバーヘッドの関数が上に凸であるときには、すべてのアクセスが均等に分布している場合

(図10(c))である。分布に偏りがある時には、オーバーヘッドの総和はこれよりも少なくなる。言い換えれば、ヘッドの平均移動距離 $L_a$ におけるオーバーヘッドが繰り返し発生する時、オーバーヘッドの総和は最悪(最大)になる。

【0066】図9の $T_d(L)$ は鋸歯状の関数であるが、これに上から包絡するような上に凸な関数で $T_d(L)$ を代用すれば上記の議論は近似的に成立する。このような関数の例を図12に示す。つまりアクセス1回あたりの最悪のオーバーヘッドは、図9のグラフにおいて $T_d(L)$ 値を距離が $L_a$ のところで読みとった値とな

る(図中の $T_{max}$ )。前述のように、これは近似であるが、図9の例のように、通常 $L_a$ 付近では $T_d(L)$ と上から包絡する関数とは一致するので、事実上誤差はないと考えてよい。また、近似は安全側(オーバーヘッドを大きく見積もる側)なので、最悪値を本来より小さく見積もる危険はない。

【0067】図2に示すステップS3では、上記式(18)で与えられる回転待ちの直線群のひとつがシーク時間の関数 $T_s(L)$ よりも上でかつなるべくそれに近いようにスキューを選択した。これによって、距離 $L_a$ 付近において $T_d(L)$ を小さくすることができ、ひいては最悪のオーバーヘッド $T_{max}$ を小さくすることができる。

【0068】図11は従来のSCANアルゴリズムにおけるオーバーヘッドを示すものである。従来のSCANアルゴリズムにおいても、オーバーヘッドの総和はアクセスが均等に分布するときに最悪となる。しかし、本実施形態とは異なり、回転待ちに対する考慮がないために、ヘッドのシークが終了した後も最悪で1回転の回転待ちが発生することを考慮しなくてはならない。このため、 $L_a$ におけるシーク時間 $T_s(L_a)$ に1回転の周期 $T_{rot}$ を加えたものが最悪のオーバーヘッドとなる。図11と図9を比較すれば明らかなように、これは本実施形態による方法と比較してはるかに大きな値となる。我々の実験では、最悪のオーバーヘッドが従来のSCANアルゴリズムと比べ、本実施形態ではほぼ半分になることが確認されている。

【0069】以上に説明したように、本実施形態に係わる記録ディスクアクセス制御装置ではスキューとギャップを適切に選ぶことで、平均のヘッド移動距離 $L_a$ におけるオーバーヘッド $T_d(L)$ を最低限に抑えることが可能であり、これによって回転待ち時間を小さくすることができる。図2にあげたフローチャートではブロックの大きさは与えられた固定の値であったが、用途によってはブロックの大きさのある程度の範囲で選択可能である。この場合はギャップ $\theta_g$ とスキュー $\theta_s$ の両方を変化させることができるので、 $L_a$ 付近でシーク時間に近付くようにより細かく直線の位置を制御することができる。

【0070】以上の方法によって、ブロック間の移動に伴うオーバーヘッドは大きく改善される。しかしブロックが大きく、複数のトラック、あるいは複数のシリンダにまたがる場合には、トラック変更にとまなう時間や隣接シリンダへの移動時間も考慮しなくてはならない。トラックの変更も隣接シリンダへの移動も必要な時間はそれぞれ一定なので、この時間を経た後にちょうどヘッドの下にデータが到着するようにトラック間あるいはシリンダ間でスキューを与えておくことにより、トラックの変更やシリンダの移動に伴ってブロック内で長い回転待ちが発生することを予防できる。このためにはブロックア

ロケータ1はすでに説明したブロック間の移動のためのスキューとギャップの他に、ブロック内をもっとも高速にアクセスするための別のスキューとギャップの組を持ち、図2におけるステップS4、S5において、1ブロックをディスク上に配置する時には後者のスキューとギャップを用いてブロックの配置を行なえばよい。後者のギャップは通常0にでき、スキューだけでトラックの変更やシリンダの移動時間を吸収できる。

【0071】次に、スケジューラ2の動作について説明する。図3はスケジューラ2のフローチャートを示すものである。ステップS11でまずディスクのヘッドをシリンダ「#0」に移動させる。次いで実際のスケジューラを行なうステップS12に移る。ステップS12では、ステップS12-1において時間的に前の（古い）ものから順にN個のアクセスリクエストをアクセスリクエストバッファから読み込む。アクセスリクエスト1個にはアクセスすべきブロックの番号と、データ転送に使用するデータバッファの先頭アドレスとが記述されている。また、個数Nはあらかじめ別の制御プログラムから与えられる定数である。

【0072】ステップS12-2においてN個のアクセスリクエストのそれぞれについてブロックマップ3を参照し、アクセスすべきブロックの物理アドレス（シリンダ番号、メディア番号、セクタ番号）を知る。次いでステップS12-3においてこれらN個のアクセスリクエストをシリンダ番号の小さい順に並べ替える。この操作によってSCANアルゴリズムのスケジューラが実現される。ステップS12-4では並べ替えられたこれらのアクセスリクエストをシリンダ番号の小さいものから順にデバイスドライバを介してディスクに送り、実際のアクセスとデータ転送を行なう。1ブロック分のアクセス指示を出したらステップS12-5で転送の終了を待ち、次いで次のアクセス指示を出す。これをN回繰り返すことで（S12-6）、N個のアクセスリクエストの処理が終了する。最後に、ステップS12-7においてN個のアクセスの処理が終了したことを制御プログラムに通知して、当該N個のアクセスリクエストに関する一連の処理は終了する。尚、このフローの最後において、アクセス要求が全て処理され、アクセス要求バッファ4が空になったかどうか判断される（S12-8）。その後、スケジューラ2はステップS12-1に戻り次のN個のアクセスリクエストを取り出して処理を続行する。もしアクセスリクエストバッファにN個のリクエストがなければ、このステップで待つ。尚、例えば、ブロックアロケータ1は、ディスク5の外側から内側の全領域にわたってギャップが一定であるようにブロックの大きさを変化させたりすることで、アクセスのリアルタイム性をさらに高めることができる。

#### 【0073】第2実施形態

以上に説明した第1実施形態のスケジューラ2は、内周

まで移動したヘッドは次のスキューの最初で最も外側のアクセスの位置まで移動していた。すなわち、図3におけるステップS12-4、S12-5で最後のアクセスを行なった時にヘッドはN個のアクセスの中では最も内側のアクセスをしており、次のN個を処理するループの最初のアクセスでは最もシリンダ番号の小さい、外側のシリンダへの移動が発生する。この移動に関しては最悪の場合再内周から再外周までの最大長のシークと、1回転の回転待ちが発生する。N個のアクセスごとにこれが発生し、この間データのアクセスはできないのでこれをスキュー全体でのオーバーヘッドの総和に加算しなくてはならない。当然、その分だけ性能は低下する。

【0074】もちろん内周から外周に向けて移動する最中にもディスクのアクセスは可能であるが、ヘッドの移動方向が逆になるので、前記式（18）における最初の項、つまりスキューの項の符号が反転する。このため外周から内周に向けての移動の場合に最適なスキューとギャップは、反対向きの移動には必ずしも適切なパラメータではなくなり、逆方向にヘッドが移動した時の性能が低下する原因となる。

【0075】以下に説明する本発明の第2の実施形態に係わる記録ディスクアクセス制御装置はこの問題に対処し、内周から外周へ戻る時にも高速なデータ転送を行なう方法を提供する。まず、ブロックアロケータ1は図13に示すように、シリンダを外から内へスキューする時に使用する斜線で示されるシリンダ50と、内から外へスキューする時に使用するシリンダ51とに分ける。図13においてFを付したシリンダ50は外から内へスキューするときに使用し、Bを付したシリンダ51は内から外へスキューする時に使用する。図13では2シリンダずつ分割しているが、これに限らず、適当な単位でシリンダを分ければよい。

【0076】前述した第1の実施形態に係わる記録ディスクアクセス制御装置と同様に、図2のステップS1に従ってスキューとギャップを求めた後の図2に示すステップS4、S5の処理を図14に示すように変更する。図14においてステップS31は第1実施形態と同様に、割り当てる物理アドレスのポインタを初期化する。次いで、ステップS32で各ブロックをディスクに割り当てていく。ステップS32はすべてのブロックについて繰り返すループであり、ステップS32-1では処理中のブロックの物理アドレスをもとにブロック全体がFの領域あるいはBの領域に属するかどうかを判定する。

【0077】ステップS32-2はこの判定結果に基づく分岐である。もし1ブロック全体がFの領域に属するならばステップS32-3～ステップS32-6を実行する。このうちステップS32-3は図2におけるステップS5-1と同じで、物理アドレス表を用いて、対応する論理セクタ番号をしらべるステップであり、またステップS32-4は図2におけるステップS5-2と同



じでブロックマップ3への記入を行なう。

【0078】図6に示した第1実施形態のブロックマップ3と比べ、本実施形態におけるブロックマップ3はそのブロックがFの領域に存在するかBの領域に存在するかを示すフラグが追加されている。この様子を図15に示す。ステップS32-5はこの部分へFと記入するステップである。ステップS32-2の分岐で1ブロックが完全にFの領域に属さない場合は、その物理アドレスへの割り当てを行わず、完全にFの領域の内側になるような物理アドレスを探す。ステップS32-6では次の物理アドレスをスキューとギャップから求め、さらにそれがどの領域に属するかを調べる。ステップS32-7では、最内周まで読み取りが行われたか否かが判断され、そうなければ、ステップS32-1へジャンプし、ふたたび判定のためのS32-2を実行する。こうして再試行を繰り返すうちに再び全体がFの領域に属するような物理アドレスが得られるので、これをブロックに割り当てる。

【0079】例えば、図15において、第1実施形態の配置方式であれば次のブロック5番は物理アドレス(1/5/8)においていたが、これではブロックの後半はシリンダ2番にまたがってしまう。シリンダ2番はBの領域なので、この領域を割り当てることはせず、次に割り当て可能なアドレスを順次探していく。物理アドレスポインタを逐次進めていき、ブロック5番に関しては物理アドレス(4/2/0)が割り当てられる。第1の実施形態のように、途中をとばさずにブロックを割り当てる方法でも、この実施例のように途中をとばしながらブロックを割り当てる方法でも、半径方向の距離(シリンダ数)とスキュー量との関係は一定に保たなくてはならないので、以上に説明したような割り当て方法を使用する。シリンダ4番の先頭(4/0/0)から使用しないのもこのためである。以上の処理を物理アドレスが最内周に到達するまで繰り返す。

【0080】ステップS33~S36は同様な処理をBの領域に対して行なうものである。Bの領域の先頭は図13に示すようにシリンダ2番なので、ステップS33で物理アドレスのポインタをこのアドレスに初期化する。次いで、スキューの符号をステップS34で反転する。Bの領域は内周から外周に向けてアクセスするため、シリンダの移動量が負になる。そこでスキューの符号もこれに対応させて反転すれば内から外へのヘッドの移動に最適なスキューとなる。

【0081】ステップS35は実際にブロックマップに書き込むステップであり、この部分はステップS32-1~S32-6と同様である。ただし、(1)ブロックマップにはブロック全体がBの領域のときにのみ書き込み、そうでなければ新しい物理アドレスで再試行する点と、(2)ブロックマップにはBを記入する点とがステップS32-1~S32-6と異なる。

【0082】最後に、ステップS36において、全てのブロックについて処理が終了したが判断され、終了していない場合には、ステップS31の処理に戻る。一方、既に全てのブロックについて処理が終了していれば、このステップで処理が停止する。

【0083】図15では、Fの領域には総計で5012ブロックを割り当てた場合の例を示している。このとき、Bの領域のブロック番号は5013から開始し再び再内周に到達するまでブロックの割り当てを繰り返す。

【0084】以上のように、本実施形態では、ヘッドがディスクにアクセスしている期間または後に、ヘッドの次のアクセス動作のスケジュールが決定される。この次のアクセスが開示されたとき、ヘッドは、そのようなヘッドの次のアクセス動作のために既に決定されたスケジュールによって決定されたスタート位置に移動される。図1に示す装置の電源を投入した後にヘッドが最初にアクセス動作をする場合に、電源投入後に最初に決定されたスケジューリングによって、ヘッドはスタート位置に移動される。

【0085】ブロックアロケータ1を以上のような構成とし、スケジューラ2はヘッドが外周から内周方向へ向かっている時にはFの領域をアクセスするリクエストのみをアクセスリクエストバッファから取り出してスケジュールし、それが完了して内周から外周へ移動するときにはBの領域をアクセスするリクエストのみをアクセスリクエストバッファから取り出してスケジュールする。これによって、ヘッドの移動方向には関わらず常に回転待ちを最小限に抑えることができる。第1の実施形態では内周に達したヘッドが外周に戻るための待ち時間が発生していたが、本第2の実施形態ではこのような待ち時間はなく、ディスクの性能が向上する。

【0086】なお、第1の実施形態において1つのブロックが大きく複数のトラックやセクタにまたがるときには、これらの時間を考慮した別のスキューを与えると効率がいよことを述べた。本第2の実施形態においても同様な工夫が可能である。本第2の実施形態においては、ヘッドが内周から外周に向かう時にはブロック内のアクセスにおいても内側から外側のシリンダへ順にヘッドが移動するため、シリンダ移動に必要な時間を得るためのスキューは、外周から内周へ向かう時とは逆向きに与えればよい。

【0087】尚、図13に示すように、シリンダ50とシリンダ51とをディスク上の最内周から最外周に分散するように設けることで、ディスクアクセスの性能をさらに向上させることができる。

【0088】

【発明の効果】以上に述べたように、本発明の記録ディスクアクセス制御方法およびその装置によれば、記録ディスクのオーバヘッド、つまりシーク時間と回転待ち時間の和を低減でき、しかもその最大値を低い値で保証で

きる。また、本発明の記録ディスクアクセス制御方法およびその装置によれば、ヘッドの移動方向にみあったスキューを領域ごとに与えることによって、ヘッドの移動方向に関わらず前述のオーバーヘッドの低減が可能である。

【図面の簡単な説明】

【図1】本発明の第1実施形態に係わる記録ディスクアクセス制御装置の構成図である。

【図2】図1に示すブロックアロケータにおける処理を説明するためのフローチャートである。

【図3】図1に示すスケジューラにおける処理を説明するためのフローチャートである。

【図4】ドライブのシーク時間とシーク距離との関係を示す図である。

【図5】ブロックアロケータによるブロックのディスク上への配置方法を説明するための図である。

【図6】ブロックマップの内容を説明するための図である。

【図7】図2に示すステップS3における処理の内容を説明するための図である。

【図8】図2に示すステップS3における処理の内容を説明するための図である。

【図9】シークと回転待ちの両方を考慮した場合におけるオーバーヘッドを説明するための図である。

【図10】ディスク上のアクセスの分布の一例を説明するための図である。

【図11】従来SCANアルゴリズムにおけるオーバーヘッドを説明するための図である。従来の方法によるオーバーヘッドの関数

【図12】鋸歯上の関数を包絡する上に凸な関数を用いた場合におけるシーク距離 $L$ とディレイとの関係を説明するための図である。

【図13】本発明の第2実施形態に係わる記録ディスクアクセス制御装置のブロックアロケータにおける処理を説明するための図である。

【図14】図13に示す記録ディスクアクセス制御装置におけるブロックアロケータの処理を説明するための図である。

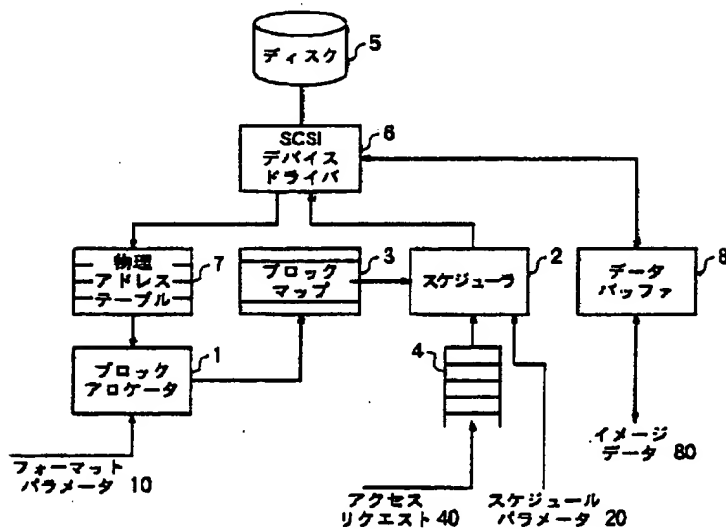
【図15】図13に示す記録ディスクアクセス制御装置におけるブロックマップの内容を説明するための図である。

【図16】従来技術を説明するための図である。

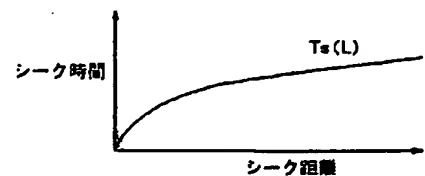
【符号の説明】

- 1… ブロックアロケータ
- 2… スケジューラ
- 3… ブロックマップ
- 4… アクセスリクエストバッファ
- 5… ディスク
- 6… SCSIデバイスドライバ
- 7… 物理アドレステーブル
- 8… データバッファ

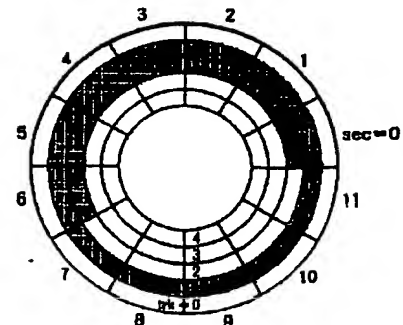
【図1】



【図4】

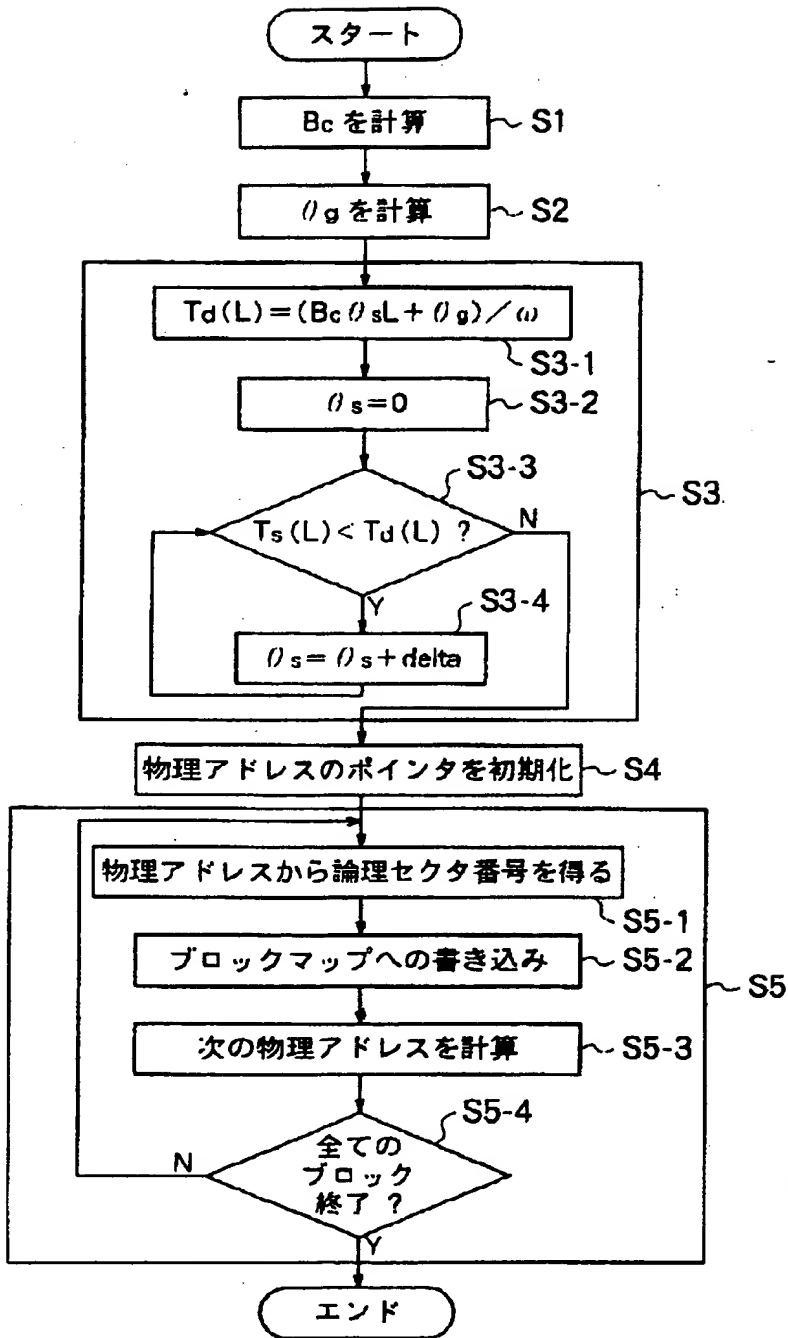


【図5】

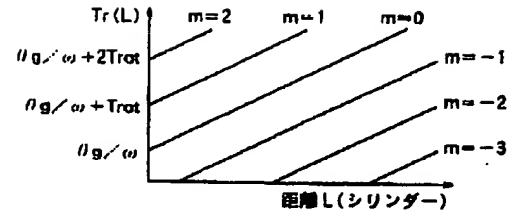




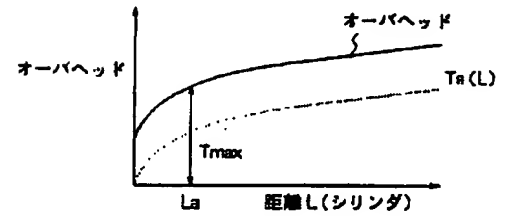
【図 2】



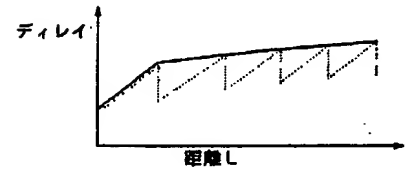
【図 8】



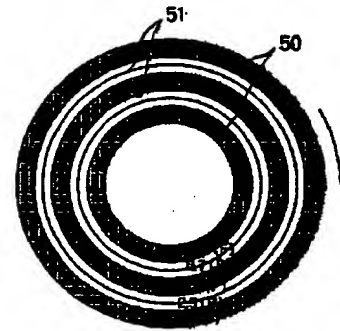
【図 11】



【図 12】

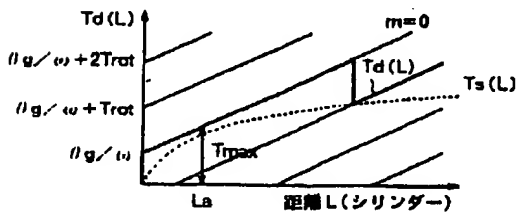


【図 13】

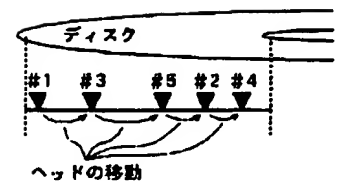
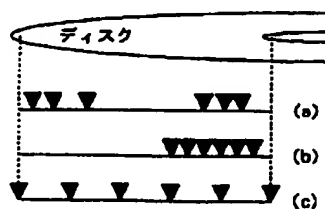


【図 16】

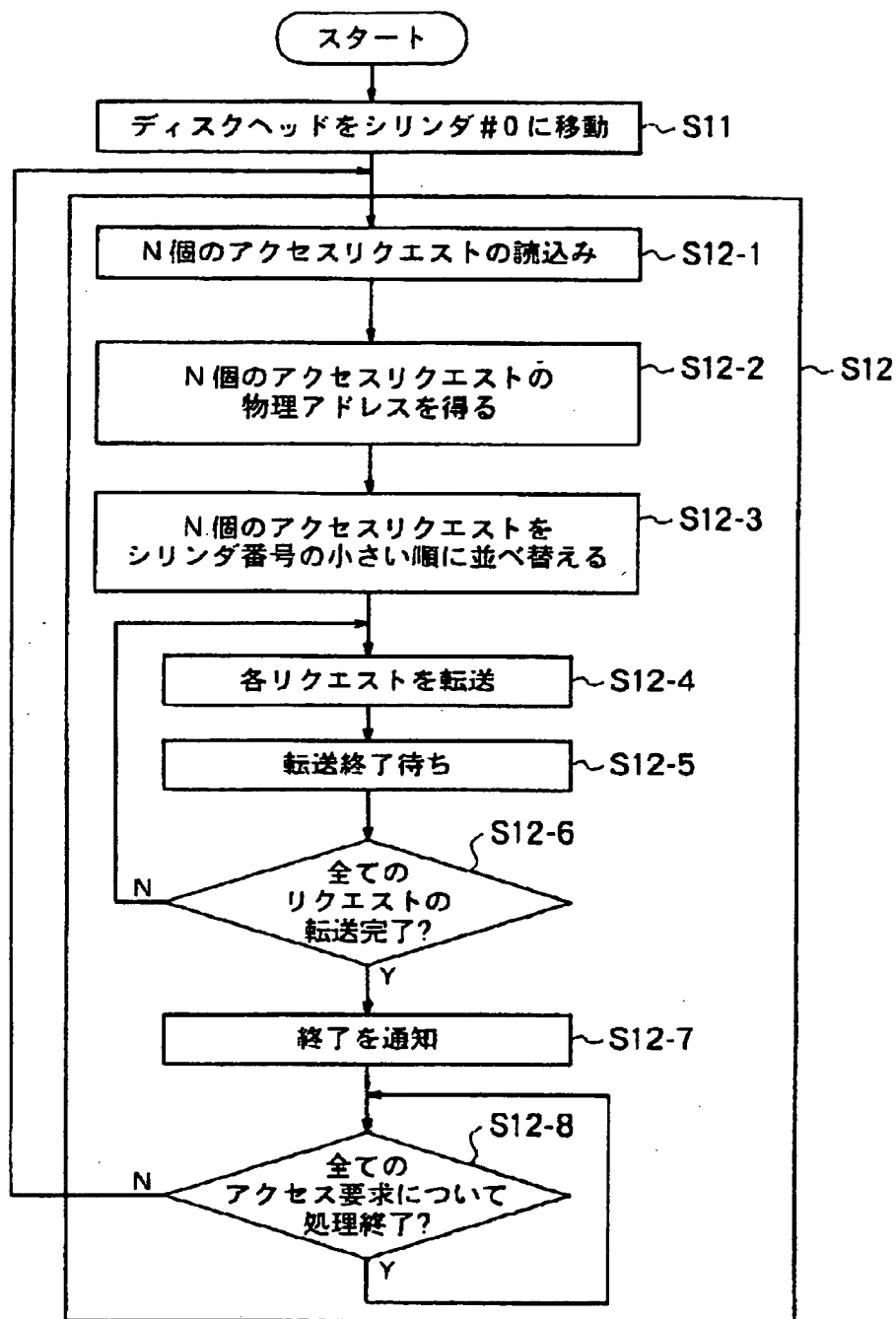
【図 9】



【図 10】



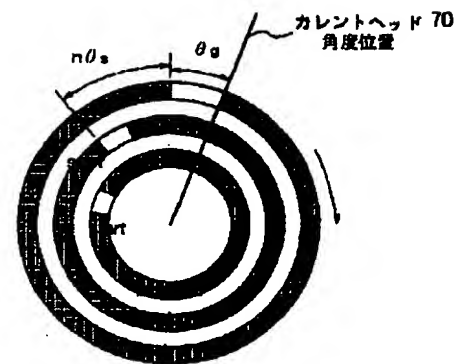
【図3】



【図6】

ブロック#	スタート 物理アドレス (cyl/Media/sec.)	スタート SCSI 論理セクタ	トータルナンバ セクタ
0	(1/0/0)	12	19
1	(2/0/8)	32	19
2	(4/0/4)	52	19
3	(8/0/0)	72	19
4	(7/0/8)	92	19
5	(9/0/4)	112	19

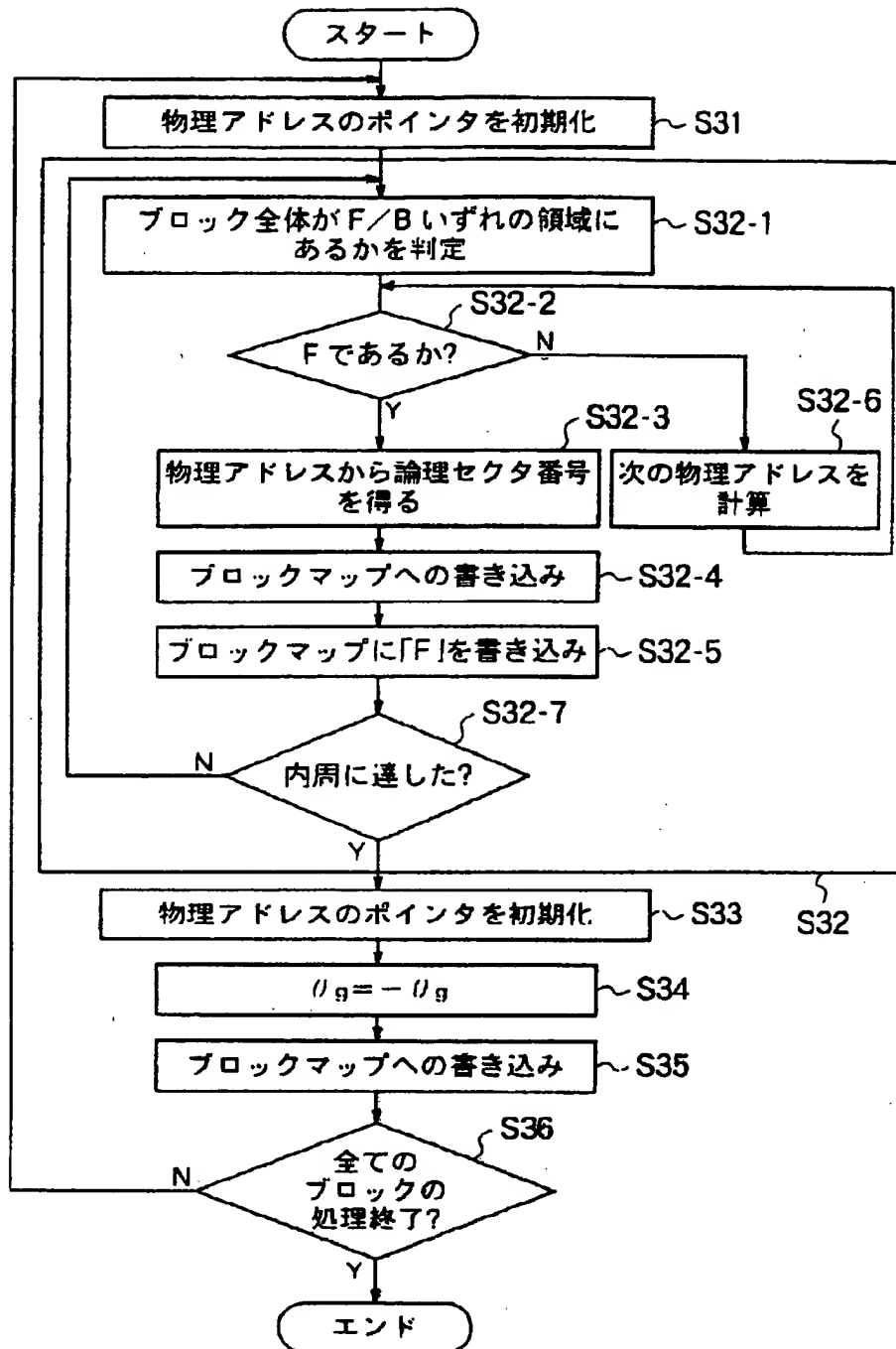
【図7】



【図15】

ブロック#	スタート 物理アドレス (cyl/Media/sec.)	スタート SCSI 論理セクタ	トータルナンバ セクタ	F/B
0	(0/0/0)	0	40	F
1	(0/2/8)	40	40	F
2	(0/5/0)	80	40	F
3	(1/0/8)	120	40	F
4	(1/3/0)	160	40	F
5	(4/2/0)	480	40	F
6	(4/4/8)	520	40	F
5013	(2/0/0)	224	40	B
5014	(2/2/8)	264	40	B
5015	(2/5/0)	304	40	B
5016	(3/0/8)	344	40	B
5017	(3/3/0)	384	40	B

【図14】



フロントページの続き

(72)発明者 塩谷 浩之

東京都品川区北品川6丁目7番35号 ソニ  
株式会社内